***Bài tập 1:***

Giao dịch bán hàng: TPS (Xử lý giao dịch theo thời gian thực: lập đơn, thanh toán, cập nhật tồn kho, in hoá đơn)

Phân tích xu hướng kinh doanh: DSS (Phân tích/khai phá dữ liệu, dự báo, what-if, phát hiện xu hướng và bất thường để hỗ trợ ra quyết định)

Bảng tổng quan hiệu suất hàng tháng dành cho CEO: EIS/ESS (Dashboard cấp lãnh đạo: KPI tổng hợp, drill-down, cảnh báo ngoại lệ, so sánh theo thời gian/kế hoạch)

***Bài tập 2:***

Dự án A: Phần mềm quản lý điểm cho trường cấp 2, yêu cầu rõ rang

Chọn; Waterfall

Vì: -Yêu cầu đã ổn định, không có sự thay đổi, phạm vi quen thuộc

-Dễ lập kế hoạch, tài liệu hóa rõ,phù hơp giai đoạn tuần tự

- Rủi ro giảm đi, ít thay đổi giữa chừng, kiểm soát chi phí và thời gian tốt

Dự án B: Ứng dụng mobile đặt lịch khám bệnh, yêu cầu linh hoạt

Chọn: Agile

Vì: Nhu cầu thị trường thay đổi nhanh, cần phản hồi người dung liên tục

Phát hành và nhận feedback sớm, ưu tiên blog linh hoạt, cải thiện trải nghiệm tốt hơn

Dự án C: Hệ thống ngân hàng điện tử có bảo mật và độ phức tạp cao

Chọn: Spiral

Vì: Rủi ro kỹ thuật và an toàn thông tin lớn, kiến trúc phức tạp, tích hợp nhiều hệ thống lõi, kiểm soát rủi ro an ninh, hiệu năng, khả năng mở rộng ngay từ đầu; giảm chi phí sửa lỗi muộn.

***Bài tập 3:***

|  |  |
| --- | --- |
| Thành phần | Mô tả |
| Con người | Khách hành(Tìm món, đặt hang, thanh toán,đánh giá)  Nhà hang(menu, nhận và chuẩn bị đơn, xác nhận món và số lượng món)  Tài xế(Nhận quốc, đến quán lấy hàng, đi giao cho khách)  CSKH(Xử lý sự cố , hoàn tiền, hỗ trợ vấn đề sản phẩm)  Quản lý hệ thống(vận hành, bảo mật,thiết lập khuyến mãi)  Đối tác thanh toán(bank) |
| Dữ liệu | Dữ liệu người dùng(Tài khoản, địa chỉ, lịch sử đơn hàng, phương thức thanh toán)  Danh mục(Món, giá, tùy chọn, hàng còn lại, giờ hoạt động)  Đơn hàng(trạng thái: tạo – xác nhận – đang chuẩn bị - đang giao – hoàn tất/hủy, phí phát sinh, khuyến mãi)  Lộ trình (GPS tài xê, vị trí quán khách)  Tài chính(Giao dịch, đối soát, phí nền tảng)  Phản hồi đánh giá |
| Quy trình | Đặt hàng: app – định vị - chọn quán – tùy chọn topping – áp vocher – thanh toán – tạo đơn  Xử lý: Nhận đơn – xác nhận/hết hàng – chuẩn bị  Giao hàng: book tài xế - nhận tại quán – giao cho khách – xác nhận đã nhận hàng  Phát sinh: hủy đơn, đổi tài xế, thất bại thanh toán, thời tiết |
| Phần mềm | Ứng dụng khách: IOS/ADR/Wed cho khách, app tài xế,app nhà hàng  Backend: Menu,Search , order,payment, …  Hạ tầng phần mềm: MySql,… |
| Phần cứng | Thiết bị liên lạc: điện thoại khách/tài xế  Thiết bị tại quán: Pos in đơn  Mạng: 4G/wifi |

***Bài tập 4:***

|  |  |
| --- | --- |
| Giai đoạn | Việc cần làm trong dự án “Ứng dụng điểm danh” |
| Planning | Xác định mục tiêu (điểm danh nhanh, chính xác, giảm gian lận); phạm vi (app GV + app/sinh viên + web admin); ràng buộc (ngân sách, thời gian, thiết bị); lập kế hoạch mốc, nguồn lực, rủi ro (mạng yếu, thiết bị cũ). |
| Analysis | |  | | --- | |  |  |  | | --- | | Thu thập yêu cầu từ Phòng đào tạo/GV/SV; use case (điểm danh theo tiết, theo tuần, học phần; báo cáo vắng/muộn; xin phép); quy tắc nghiệp vụ (tỉ lệ chuyên cần, bù vắng); mô hình dữ liệu khái niệm (Lớp, Môn, Buổi học, Sinh viên, Bản ghi điểm danh); phi chức năng (bảo mật, hiệu năng, offline). | |
| Design | |  | | --- | |  |  |  | | --- | | Thiết kế kiến trúc (mobile + API + DB; cân nhắc microservice nhỏ: attendance, course, auth); luồng điểm danh (QR/NFC/GPS + mã phiên học); thiết kế DB (ERD, chỉ mục); API spec (REST/GraphQL); UI/UX wireframe; thiết kế bảo mật (SSO/OAuth2, mã hóa, chống “proxy điểm danh”). | |
| Implementation | |  | | --- | |  |  |  | | --- | | Lập trình: app GV (tạo phiên, quét/duyệt), app SV (quét QR, xác thực vị trí), web admin (quản trị môn/lớp/báo cáo); backend (Auth, Attendance, Course, Notification), tích hợp Push/SMS/Email; đồng bộ offline → online; viết migration/seeder dữ liệu. | |
| Testing | |  | | --- | |  |  |  | | --- | | Unit test, API/integration test; test hiệu năng giờ cao điểm; test bảo mật (role, chặn giả mạo vị trí/QR chụp lại); UAT với 1–2 khoa; kiểm thử trên thiết bị đa dạng (Android/iOS đời cũ, mạng yếu), kịch bản ngoại lệ (mất mạng, trùng điểm danh). | |
| Desloyment & Maintenance | Triển khai staging → production (CI/CD), phát hành app store/internal MDM; giám sát (logs, uptime, lỗi), sao lưu & khôi phục; hỗ trợ người dùng, vá lỗi; tối ưu KPI (tốc độ điểm danh, tỉ lệ lỗi), phát triển tính năng mới (xin phép vắng, cảnh báo chuyên cần), cập nhật bảo mật/OS định kỳ. |

***Bài tập 5:***

**3 giai đoạn đầu của SDLC**

1. Planning:

+Mục tiêu: GV tao phiên học và hiển thị QR tự động, sinh viên quét điểm danh, Phòng ĐT xem được thống kê lớp/môn/buổi

+Đối tượng: Giảng viên, Sinh viên, Phòng đào tạo, Admin

+Lộ trình: Sprint 0(QR), Sprint(tạo phiên, quét), Sprint3 (Báo cáo,phân quyền)

+Rủi ro: Mạng yếu, lộ QR, giờ cao điểm

+KPI: p95 check in<2-3s, giảm gian lận>80%, uptime>=99% trong giờ học

1. Requirement Analysis:

Case chính:

+**GV:** tạo/bật-tắt phiên, hiển thị QR, xem/sửa danh sách, khóa phiên.

+**SV:** đăng nhập SSO, xem lịch, quét QR, xem lịch sử.

+**PĐT:** dashboard, lọc/kết xuất CSV, cấu hình quy tắc chuyên cần.

+**Admin:** quản trị người dùng, môn/lớp, tham số hệ thống.

1. System Design:

Tổng quan:

+**Clients:** App/Web SV (quét QR, gửi GPS/BSSID), Web/App GV (tạo phiên, hiển thị QR), Web PĐT/Admin (dashboard/cấu hình).

+Bảo mật & chống gian lận: QR **ngắn hạn + one-time**, geo/BSSID, device-fingerprint, rate-limit, phát hiện trùng tọa độ/thiết bị; RBAC & audit đầy đủ.

***Bài tập 6:***

|  |  |
| --- | --- |
| Tình huống | Sơ đồ UML phù hợp |
| 1. Mô tả chức năng người dung có thể thực hiện trên ứng dụng học tiếng Anh | |  | | --- | | Use Case Diagram |  |  | | --- | |  | |
| 1. Mô tả lớp NguoiDung, khoaHoc, baiHoc và quan hệ giữa chúng | |  | | --- | | Class Diagram |  |  | | --- | |  | |
| 1. Mô tả luồng học viên bắt đầu -> vào học ->làm bài ->hoàn thành | |  | | --- | | Activity Diagram |  |  | | --- | |  | |
| 1. Mô tả cách hệ thống triển khai trên các máy chủ, thiết bị | |  | | --- | | Deployment Diagram |  |  | | --- | |  | |
| 1. Mô tả thứ tự tương tác giữa học viên và hệ thống khi nộp bài | |  | | --- | | Sequence Diagram |  |  | | --- | |  | |

***Bài tập 7:***

6 giai đoạn SDLC cho hệ thồng đăng ký tiêm online:

* Planning: Xác định mục tiêu, phạm vi (Đăng ký – xếp lịch – theo dõi), mốc thời gian, rủi ro
* Requirement: Thu thập yêu cầu các bên, đăng ký, xác nhận, xếp lịch, tra cứu, quy tắc, ưu tiên
* System Design: Kiến trúc Wed/Mobile+Api+DB, thiết kế dữ liệu, bảo mật
* Implementation: Agile, xây dựng các modul
* Testing: Kiểm tra hiệu năng giờ cao điểm, kiểm bảo mật
* Deployment: Mở rộng, hỗ trợ người dung, cải tiến UI và bảo mật

***Bài tập 8:***

* Tác nhân và chức năng:

+ Học viên: Đăng ký tài khoản,vào học,làm bài thi, xem điểm, nhận kết quả, hỏi đáp

+ Giảng viên: Tạo/chỉnh sửa bài học(Nội dung bài tập quiz), quản lý lớp, chấm điểm, xuất bảng điểm

+Admin: Quản trị người dùng, khóa học lịch học, cấu hình học phí, phê duyệt nội dung, báo cáo và thống kê, phân quyền, giám sát hệ thống

+ Kế toán: Đối soát thanh toán, hóc đơn, hoàn tiền

+ Hệ thống tích hợp: Cổng thanh toán

* Phân loại hệ thống thông tin:

+ TPS: Đăng kí tài khoản, thanh toán điểm danh, nộp bài, chấm điểm

+ MIS: Báo cáo sĩ số, tỉ lệ hoàn thành, doanh thu theo tháng

+DSS: Gợi ý lộ trình học , phát hiện rủi ro bỏ học, mở thêm lớp học

+EIS: Trang tổng quan doanh thu, tang trưởng học viên, cảnh báo

* Mô hình phát triển phầm mềm đề xuất:

**Agile (Scrum) + Incremental**, kèm **spike** kỹ thuật cho video/quiz ở quy mô lớn.

Nhu cầu thay đổi nhanh (UX học, hình thức kiểm tra, khuyến mãi); cần phát hành sprints ngắn để nhận phản hồi học viên/giảng viên; tích hợp nhiều dịch vụ (thanh toán, CDN) cần thử nghiệm sớm.

**Cách làm:** Sprint 2 tuần; ưu tiên các increment:

* 1. Đăng ký/ghi danh + thanh toán (TPS)
  2. Học liệu/quiz/chấm điểm
  3. Báo cáo quản lý (MIS)
  4. Gợi ý học/alert rủi ro (DSS)
  5. Dashboard lãnh đạo (EIS).CI/CD, feature flags, đo KPI (tỷ lệ hoàn thành bài, p95 latency, tỉ lệ lỗi thanh toán).

***Bài tập 9:***

**1. Actors & chức năng**

-Khách hàng: tạo đơn, thanh toán/COD, theo dõi trạng thái/ETA, đổi địa chỉ/hẹn lại, khiếu nại.

-Shipper: nhận tuyến, quét mã vận đơn, cập nhật trạng thái (lấy/giao/thất bại + lý do), chụp PoD.

-Điều phối: gán đơn, tối ưu lộ trình, xử lý ngoại lệ (giao lại/chuyển kho/hoàn).

-Kế toán: đối soát COD/online, công nợ, hoá đơn.

-Quản lý vận hành: theo dõi OTD, backlog theo hub/khu vực/nhân viên, báo cáo tuần/tháng.

-CEO: tổng quan doanh thu–đơn, vùng hoạt động/heatmap, OTD theo khu vực, cảnh báo.

**2.Phân loại HTTT**

-TPS: tạo đơn, tính phí/in nhãn, quét mã, cập nhật trạng thái, tracking realtime.

-MIS: báo cáo vận hành ngày/tuần, tồn đọng theo khu vực, đối soát COD.

-DSS: tối ưu điều tuyến, dự báo backlog/nhu cầu, phân bổ shipper “what-if”.

-EIS: dashboard CEO tóm lược (doanh thu, OTD, bản đồ vùng hoạt động, cảnh báo).

**3.Mô hình phát triển đề xuất**

-Agile (Scrum) + Incremental, kèm spikes kỹ thuật cho điều tuyến/dự báo.

-Lý do: nghiệp vụ thay đổi nhanh, nhiều tích hợp (bản đồ/thanh toán), cần phát hành sớm–liên tục để lấy feedback hiện trường.

-Lộ trình: V1 tập trung TPS + MIS cơ bản → V2 bổ sung DSS + hoàn thiện EIS. CI/CD, feature flags, p95 API 300–500ms, cập nhật trạng thái idempotent.

**4. UML dùng khi thiết kế**

-Use Case Diagram: actors và ca sử dụng chính (tạo đơn, cập nhật trạng thái, điều tuyến, báo cáo, dashboard).

-Activity Diagram: luồng end-to-end từ tạo đơn → nhập kho → phân tuyến → giao → xử lý ngoại lệ → đối soát.

-Sequence Diagram: tương tác khi shipper cập nhật trạng thái (App → API GW → Order/Tracking → Event Bus → Notification/Analytics).

-Class Diagram (hoặc Component): mô hình miền Order, Shipment, RoutePlan, TrackingEvent, Hub, Customer, Shipper, CODSettlement và quan hệ

***Bài tập 10:***

**1.Actors & chức năng**

**-Khách hàng:** tạo đơn, thanh toán/COD, theo dõi trạng thái/ETA, đổi lịch/địa chỉ, đánh giá.

**-Shipper:** nhận tuyến, quét mã vận đơn, cập nhật trạng thái (lấy/giao/thất bại + lý do), chụp PoD.

**-Điều phối:** gán đơn/tuyến, tối ưu lộ trình, xử lý ngoại lệ (giao lại/chuyển kho/hoàn).

**-Quản lý vận hành:** theo dõi OTD, backlog theo khu vực/hub/nhân viên, báo cáo tuần/tháng.

**-Kế toán (tuỳ chọn):** đối soát COD/online, công nợ/hóa đơn.

**-CEO:** tổng quan doanh thu–đơn, vùng hoạt động/heatmap, tỷ lệ giao đúng hạn, cảnh báo.

**2.Phân loại HTTT**

**-TPS:** tạo đơn, tính phí/in nhãn, quét mã, cập nhật trạng thái, tracking realtime.

**-MIS:** báo cáo vận hành ngày/tuần/tháng, tồn đọng theo khu vực, đối soát COD.

**-DSS:** tối ưu điều tuyến (VRP), dự báo backlog/năng lực, mô phỏng “what-if” phân bổ shipper.

**-EIS:** dashboard lãnh đạo (doanh thu, OTD, bản đồ vùng hoạt động, cảnh báo SLA) với drill-down.

**3.Mô hình phát triển phần mềm**

**-Agile (Scrum) + Incremental + spikes kỹ thuật** cho điều tuyến/dự báo.  
*Lý do:* nghiệp vụ thay đổi nhanh, nhiều tích hợp (map/payment/notify), cần phát hành sớm để lấy feedback hiện trường.  
*Lộ trình:* **V1**: TPS + MIS cơ bản → **V2**: DSS + EIS; CI/CD, feature flags, cập nhật trạng thái **idempotent**, p95 API 300–500 ms.

**4.UML đề xuất**

**-Use Case Diagram:** phạm vi & actors/ca sử dụng chính (tạo đơn, theo dõi, cập nhật, điều tuyến, báo cáo, dashboard).

**-Activity Diagram:** luồng end-to-end từ tạo đơn → nhập kho → phân tuyến → giao → ngoại lệ → đối soát.

**-Sequence Diagram:** thứ tự tương tác khi shipper cập nhật trạng thái (App → API GW → Order/Tracking → Event Bus → Notification/Analytics).

**-Class Diagram** *(hoặc Component)*: mô hình miền & quan hệ (Order, Shipment, RoutePlan, TrackingEvent, Hub, Customer, Shipper, CODSettlement).